Volume xxx, (1998) number yyy pp. 000–000

Surface Reconstruction with Anisotropic Density-Scaled Alpha Shapes

Marek Teichmann and Michael Capps

(marekt, capps)@graphics.lcs.mit.edu
Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts

Abstract

Generation of a three-dimensional model from an unorganized set of points is an active area of research in computer graphics. Alpha shapes can be employed to construct a surface which most closely reflects the object described by the points. However, no α-shape, for any value of α, can properly detail discontinuous regions of a model. We introduce herein two methods of improving the results of reconstruction using α-shapes: density-scaling, which modulates the value of α depending on the density of points in a region; and anisotropic shaping, which modulates the form of the α-ball based on point normals. We give experimental results that show the successes and limitations of our method.

1. Introduction

Generation of a three-dimensional model from an unorganized set of points is an active area of research. Such point-sets come from a number of common sources, such as range data from three-dimensional scanning hardware, implicit surface sampling [11], and medical imaging.

The notion of α-shapes provides an elegant mathematical framework for extracting the geometric structure of a set of points in three dimensions. In particular, α-shapes can be used to reconstruct a polygonal surface which approximates an input set of surface point-samples [3]. It uses distances between points to decide which input points to connect with triangles. Briefly, the α-shape is a set of triangles and tetrahedra that is a subset of the Delaunay triangulation of the input point set. The Delaunay triangulation has the property that it connects points to their closest neighbours, but produces a complete tetrahedrization of space. Since we wish to obtain a surface, we need to carefully select a subset of the triangles in the triangulation. One elegant method for doing this is to define a *forbidden region* around each potential triangle and retain only those triangles whose forbidden region is empty of all other points.

The theory of α-shapes provides such a region. For triangles not on the convex hull of the point set, it is the smallest sphere circumscribing the triangle. A triangle is then in the α-shape if the radius of this sphere is at most α.

While this definition gives good results for point sets of roughly uniform density with large separation between surfaces, this definition is clearly not optimal for non-uniform point sets, or surface which are relatively closer than their sampling density. We therefore propose two extensions to the definition of α-shapes to alleviate these problems and allow reconstruction of a much larger class of point sets.

- **Anisotropic scaling**: we allow the spherical forbidden region to vary in shape, and change the triangulation accordingly.
- **Density scaling**: we vary the value of α depending on the local point density.

For anisotropic scaling, we assume that we are given normal information about the points; if this information is not available, it can be approximated using a least-squares technique such as in [5]. The forbidden region becomes an ellipsoid whose axes and eccentricity is determined according to the local point normal information. In essence we vary the local metric tensor according to the point normals. The underlying assumption here is that the field of normals is sufficiently smooth. The Delaunay triangulation must be modified to take into account this extension and we give an incremental algorithm to retriangulate the input point set based upon the new metric tensor.

Varying scale as a factor of local point-density allows us to differentiate areas of the point set with different densities and avoid connecting triangles between such areas.

As in the case of α-shapes, these methods are meant to be used in an interactive setting. For each of the above extensions, we provide a user specified parameter which allows the user to vary the impact of their effects interactively. We emphasize that a significant contribution of this work is the ease with which it can be incorporated into α-shape or triangulation frameworks.

The remainder of this paper is organized as follows: in the next section, we describe Delaunay triangulation and α-shapes in more detail, and explain how discontinuities in the model can affect results. In Section 3, we outline our method for generating anisotropy, i.e. (non-uniform shape) ellipsoidal forbidden regions. Here we take advantage of point-normal information to detect certain discontinutities. Section 4 explains the density-scaling method, which varies the size of the α-ball throughout the model. In the remainder of the sections, we discuss our implemention of these methods and their results, and give a sketch of what future directions we see for this topic.

2. Previous Work with Alpha-Shapes

In this section we define alpha shapes and related concepts. We show how, in conjunction with a Delaunay triangulation, alpha shapes can be used to generate the desired three-dimensional form from a set of points. We also demonstrate the inherent weaknesses in that method.

2.1. Delaunay Triangulation

A set of points P defines a unique triangulation known as the Delaunay triangulation. A standard algorithm for its construction is incremental construction using Lawson's flip method [3]; a current triangulation is maintained (initially a tetrahedron), and points are inserted into the triangulation one by one. At each step, the triangulation is modified to maintain the Delaunay property: the circumscribing sphere of each tetrahedron in the triangulation cannot contain any of the input points. To maintain this property, those tetrahedra which fail a local test are *flipped* such that the new edge is part of the Delaunay triangulation.

We first describe the test and flip process in two dimensions for clarity, and then show how it is easily extended to three dimensions. We will use this flipping step in our retriangulation step, when we take normals into account.

2.1.1. Flipping in Two Dimensions

For each edge $\overline{pq}$ in an arbitrary triangulation T, we check if it is *locally Delaunay.* An edge is locally Delaunay if it is either on the boundary of the convex hull, or, if not, we find the two triangles incident to $\overline{pq}$, $\triangle pqr$ and $\triangle pqs$. If the circle circumscribed by points pqr contains point s, or if the circle circumscribed by pqs contains r, then this edge is not locally Delaunay. However, only a simple change is necessary: we flip the edge to create the other two triangles possible with points $pqrs$ (see Figure 1).

This new edge is necessarily locally Delaunay. We test each edge in this fashion; when all edges in T' are locally Delaunay, it is a Delaunay triangulation.

2.1.2. Flipping in Three Dimensions

For three dimensions, our triangulation is a set of tetrahedra over the points in P forming a simplicial complex. A set of 5 points in convex positions can have only two possible triangulations, as illustrated in Figure 1. One of them is guaranteed to satisfy the Delaunay property as described above: the circumscribing spheres of the tetrahedra involved contain no points in their interior. If one of the tetrahedra does not have this property, we can *flip* to the other triangulation. It turns out that if this procedure is applied to the tetrahedra adjacent to a newly inserted point in an incrementally constructed triangulation, we are guaranteed to obtain a Delaunay triangulation.

2.2. Alpha Shapes

The Delaunay triangulation of our point set triangulates the convex hull of the point set, and as such is not suitable for reconstruction. Instead, the α-shape of a point set is the set of triangles and tetrahedra (here for simplicity we only consider the triangles), taken from the triangles forming the Delaunay triangulation, that satisfy an additional constraint: we call it the α-test [3]. Let α be a non-negative real number. The α-test is closely related to the Delaunay property test.

Given the triangle t not on the convex hull and the points of the two adjacent tetrahedra p, q, we see if those points are within the circumsphere of t. If not, and the radius of the sphere is less than α, we accept t. If so, we find the smallest sphere circumscribing t

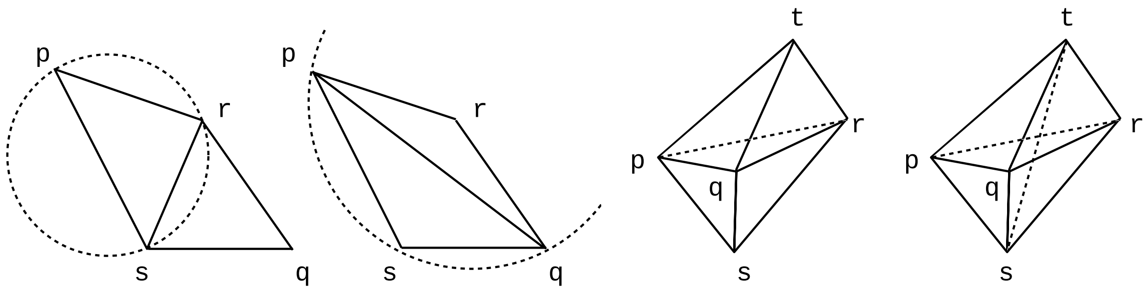

Figure 1: *Edge flipping for Delaunay triangulations. The first figures show a triangle flip in 2-D; the last show the possible tetrahedrizations of five points.*

and one of p, q. If the radius of that sphere is smaller than α, we accept.

All triangles that are rejected are not in the α-shape, so what remains is a subset of the Delaunay triangulation which, after an adjustment of α, should more closely follow the topology of the point set. We can see in Figure 2 that for very large values of α the α-shape is the convex hull; this is as we would expect, since all spheres circumscribed by points in the figure would be smaller than the α-sphere. As α approaches 0, the α-shape is exactly the point-set; since alpha is smaller than all circumscribed spheres, no triangle passes the α-test. The figure also shows a well-chosen α that yields the desired result, for a reasonably uniformly sampled surface.

2.3. Limitations of Alpha Shapes

Certain surface discontinuities and surface arrangements are not properly detected by α-shapes; for these, there exists no value of α that includes all desired triangles and deletes all undesired triangles. The following figures illustrate just a few of these situations:

- *interstice*: Figure 3 shows a break in the surface. A standard α-shape has no way to tell between surface points and the points marking the edge of the interstice, so it covers the interstice.
- *neighbor*: The figure shows two separate objects whose surfaces are near to each other. Again, an α-shape passes triangles that connect points on both objects.
- *joint*: The figure also illustrates a discontinuity where there is a sharp turn, or joint. α-shapes often give a "webbed-foot" appearance at such joints, since they improperly connect the adjacent surfaces.

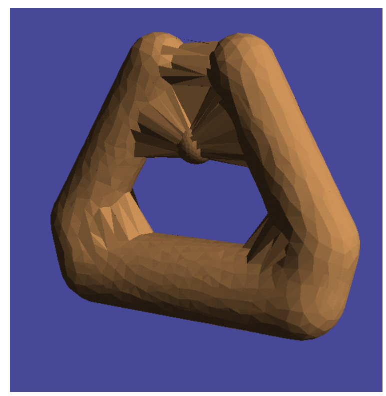

Figure 3: *α-shape demonstrating failure between surfaces and at joints and interstices*

2.4. Previous Work

The concept of α-shapes has previously been used for reconstruction of smooth surfaces [1].

Anisotropic Delaunay triangulations have been explored previously, in the application area of two-dimensional mesh generation. Bossen and Heckbert [2] apply a 2×2 metric tensor to quantify desired mesh element size as a function of position. A modified Delaunay criterion, taking this anisotropy into account, generates unstructured meshes for complex domains. They suggest extension to three-dimensions is appropriate, but that tetrahedrization is topologically much more complex than similar triangulation.

Our α-shape techniques can be used for the polygonization of implicit surfaces by reconstructing the surfaces from triangulated point samples. The direct polygonization of implicit surfaces is an entirely different thrust of research; current investigation into using

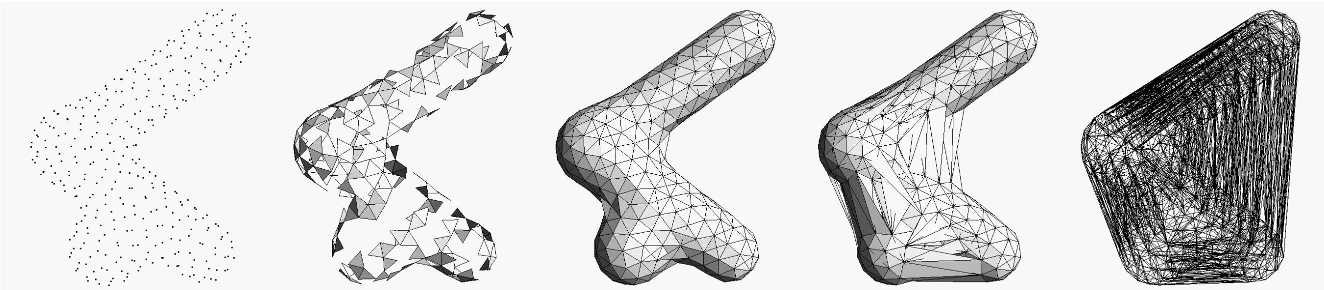

Figure 2: *α-shapes: a single point set with* $\alpha = \{0, .19, .25, .75, \infty\}$

Morse and catastrophe theory for ensure topological consistency has provided successful, though complex to implement [10].

Other techniques involve the construction of a *signed distance function*, then reconstructing the surface by finding the zero set of the function using standard polygonization techniques [7, 8]. Hoppe *et al.*, for example, assume that the points are sufficiently uniformly distributed [5].

3. Anisotropic Alpha Shapes

We claim that triangulation of a set of points can be improved significantly with anisotropic α-shapes, in which the sphere used in the α-test is deformed as necessitated by local properties. The interstice discontinuity mentioned above is an excellent example of a situation whose handling can be substantially improved by anisotropic α-shapes. We would like for triangles spanning the interstice to fail the test and be deleted from the α-shape; to do this we *compress* the α-test sphere, or α-ball, along an axis perpendicular to a local plane separating the interstice.

This procedure amounts to elongating space along that direction, which corresponds closely to the general direction of the normals in the neighbourhood of the points involved. When such surface-normal information is available at each point, we use it to modify the shape of the α-ball, and therefore the local metric tensor.

Formally, anisotropy is specified by the following parameters: the 3 radii r_i of the ellipsoid defining the metric, and the rotation of this ellipoid. The corresponding metric tensor is then

$$M = RSR^T$$

where R is a rotation matrix, and S is a diagonal matrix with $1/r_i$ on the diagonal. In our case, we will let $r_1 = r_2 = r_3$. This value will be related to the compression factor as determined by an interactive user input and the local normal correlation. Assuming the metric tensor is locally almost constant, the distance between two points x and y is then approximated by

$$d(x, y) = \sqrt{(x - y)^T M (x - y)}.$$

Measuring this distance is equivalent taking the usual Euclidean distance in a normalized space, elongated such that the ellipsoid becomes a sphere.

For instance, triangles spanning an interstice will generally contain points whose normals are nearly colinear. Thus to avoid connecting points on “opposite” surfaces, we artificially stretch space to make them appear farther apart.

In Figure 4A, we show a triangle consisting of points in two separate surfaces, in an interstice discontinuity. The original α-ball, actually an α-circle in this two-dimensional example, is easily larger than the triangle's circumscribing circle. Given the normal information for each point, and the resultant normal for the triangle d, we can compress the α-circle (Figure 4B) in a manner which will fail this interstice triangle. Or equally, we can stretch the intervening space (Figure 4C) and test against the original α-circle, also deleting the triangle as desired.

3.1. Computing the Local Anisotropy

In our system, anisotropy is determined on a triangle by triangle basis. When given three points, we obtain (possibly an approximation of) the corresponding normals, say n_1, n_2, n_3, and perform the following computation to attempt at identifying the local normal direction.

Over the eight possible sign combinations, we compute the vector that maximizes the value of n where

$$n = \max_{(s_1, s_2, s_3) \in \{(\pm 1, \pm 1, \pm 1)\}} \| s_1 n_1 + s_2 n_2 + s_3 n_3 \|$$

This vector is then normalized giving a local normal direction d. A user input parameter, which we call τ,

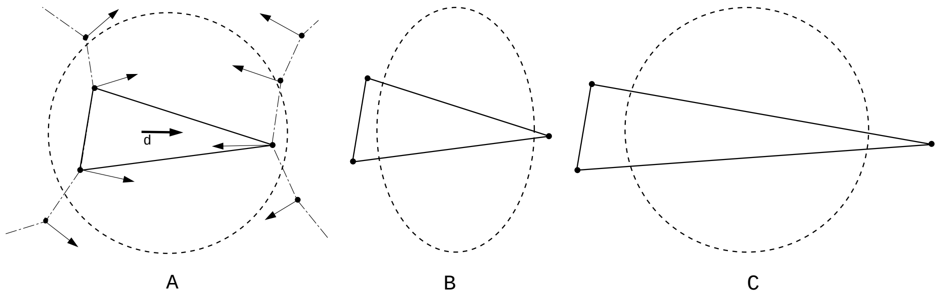

Figure 4: *Anisotropic α-test. 4A: An interstice triangle which would pass given the current α. 4B: The compressed α-balls now fails the triangle. 4C: Equally, the expanded points now fail for the original α-ball.*

is used to determine the amount of scaling the sphere undergoes in that direction. In other words the rotation matrices R align the major ellipsoid axis with d. The method above captures well the local normal direction information if the normals are sufficiently well aligned. We found that averaging the normals and using the resulting magnitude as an additional multiplicative factor produced worse results.

3.2. Retriangulating

The new α-ellipsoid is used in the α-test; the triangle's points are circumscribed to a similar ellipsoid and compared to the α-ellipsoid. In practice, it is much easier to compress the points along the d direction by τ and use the regular α-test. We call this the anisotropic α-test.

A fundamental contribution of this work is incremental retriangulation based on a user-specified factor, τ, for the influence of the anisotropy. The local normal direction d is multiplied by τ, so the user has direct control of the anisotropy; $\tau = 1$ creates an anisotropic α-test equal to the standard α-test. The user modulates τ and α to better find the desired triangulation of the input point set.

To retriangulate, we first begin with the current triangulation $T_{\alpha,\tau}$, which naturally begins as the Delaunay triangulation which is $T_{\alpha,1}$. We then apply our anisotropic α-test to each triangle in $T_{\alpha,\tau}$, and flip the triangle if it fails. Newly created triangles are added to the work queue and tested normally. This allows the possibility of a flipping cycle, which can in turn result in infinite loops of retriangulation; to avoid this, we save a set of flipped triangles and do not test the same triangle more than once.

In practice, the user adjusts τ in small increments, which causes relatively few flips.

4. Density-scaled Alpha

4.1. Density Determination

Each point p in a general set of points P has a local point density property, defined as

$$\delta(p) = \sum_{q \in P} 1 - \frac{d(q,p)}{\lambda} \qquad \forall q \text{ such that } d(q,p) < \lambda,$$

where λ is the constant radius of the local neighborhood and $d(x,y)$ is the Euclidean distance function. When $\delta(p) > \sum_{q \in P} \frac{\delta(q)}{|P|}$ (when local density is more than average), we know certain characteristics about the model region surrounding p, depending on the point distribution.

In a point set generated with uniform surface distribution, such high-density areas are necessarily those where two separate surfaces are in close proximity. This can, as expected, occur at interstice discontinuities or with two neighboring but separate objects. High densities also are observed on the inside of joint discontinuities where the surface is bent back on itself. In point sets with non-uniform distribution, higher densities are found where model description requires it–such as curved surfaces, and to a lesser degree, joints. It is also possible for separate objects in a model to have different point density, in which case δ can be used as to determine which object a point is associated with.

4.2. Scaling Algorithm

After computing density information for each point, we utilize this data when computing the α-shape. The points and circumsphere are determined as normal, but they are compared to a scaled α-ball.

This new α' will be identical to α except when $\delta(t)$ is greater than one, in which case $\alpha' = \alpha/\delta(t)^\sigma$. The σ value is a factor that is adjusted interactively by the user, similar to α, when generating an α-shape. Let μ

be the global average density, over the entire point set. $\delta(t)$ represents the density of the $\triangle abc$ being tested, which is computed in one of the following ways:

- average density: $\delta(t) = \frac{\delta(a)+\delta(b)+\delta(c)}{3\times\mu}$
- maximum density: $\delta(t) = \frac{\max(\delta(a),\delta(b),\delta(c))}{\mu}$
- maximum density difference: $\delta(t) = \frac{\max(|\delta(a)-\delta(b)|,|\delta(a)-\delta(c)|,|\delta(b)-\delta(c)|)}{\mu}$
- max density ratio: $\delta(t) = \max(|\frac{\delta(i)}{\delta(j)}|)\ \forall i,j \in \{a,b,c\},\ i \neq j$

Using the average or maximum density reduces the size of the α-ball in areas where point density is high, as desired. Using the maximum point density is more effective in deleting triangles that connect high-density points to low-density points. Note that α is not modified when $\delta(t)$ is less than the μ, since that has the undesired effect of making it easier for very large triangles on the convex hull to pass the α-test and remain in the α-shape.

Setting a triangle's density to be the largest density difference in its points causes triangles which connect high-density and low-density areas to be removed from the α-shape. This is most effective in models where separate objects have different average point densities. It can give undesirable effects, however, in non-uniformly distributed point-sets; it tends to delete triangles connecting complex and simple areas of the model such as straight edges and curved joints.

5. Implementation and Results

5.1. Implementation

To implement our α-shapes, we used as a basis the Detri 2.2 package written by Ernst Mucke [6]. It employs a variant of the randomized incremental-flip algorithm due to Edelsbrunner and Shah [4]. The time complexity of the code is roughly proportional to the number of triangles in the final triangulation. In the worst case, this is quadratic in the number of input points, but for most cases it is closer to linear. The method requires exact (long-integer) arithmetic. The resulting slow-down due to the lack of adequate hardware support is reasonably compensated by a well tuned long-integer package. Though exact arithmetic is slower than floating-point arithmetic, it is necessary to ensure a robust solution, and robustly deal with point sets which might be highly degenerate. The above package is used for the precomputation step which computes the Delaunay triangulation of the point set. After this step, we use a modified version of the Delaunay flip present in the package for the retriangualation when τ changes.

Changes in the other parameters, α and σ do not require retriangulation. To this package, we added our own floating point alpha-testing code, as well as a visualization interface. The main data structure is a *triangle-edge* data structure, a generalization of the classical winged-edge data structure for maintaining planar triangulations [9].

To allow the user to change α at an interactive rate, and thereby find a proper subset of the triangulation, we precompute the radius of the appropriate circumscribed sphere for each triangle. In this way, each change of α requires only a single comparison for each triangle to determine if it should be displayed.

The table in Figure 6 shows the times for the steps in computing an α-shape. For example, the α-shape for a set of 360 points is computed at roughly 20 Hz; the Delaunay triangulation with anisotropic tensors requires 141 milliseconds, and α-precomputation for that triangulation requires about one second. We found that computing α-shapes with density-scaling requires only approximately 25% more time than usual. For all measures, computation scales linearly with the number of points.

5.2. Results

We see in the first plate of Figure 5 that density-scaling α deletes all triangles that connect the higher-density ball to the lower-density cylinders nearby. In the second plate, we add normal-induced anisotropy which eliminates the unwanted triangles across the interstice. While the problem of joint discontinuties has not been solved, anisotropic α-shapes do offer a marked improvement over conventional α-shape. Generally, the joint "webbing" will occur up to the point sampling frequency.

The two methods combine well in this example. The anisotropy method does remove triangles connecting the surfaces of the sphere and the cylinders where they face each other. However, it has no effect on triangles that would ordinarily be on the convex hull, i.e. those connecting points with similar (outward) normals. Density-scaling works well in exactly that situation, as seen in the figure.

6. Conclusions and Future work

Anisotropic α-shapes appear to be a promising new method for reconstructing triangulated surfaces from point sets, which have or from which one can extract local normal information. They are clear superior in the handling of "difficult" point sets that regular α-shapes. However, they rely on user input and it would be interesting to algorithmically determine the settings for the user specified parameters.

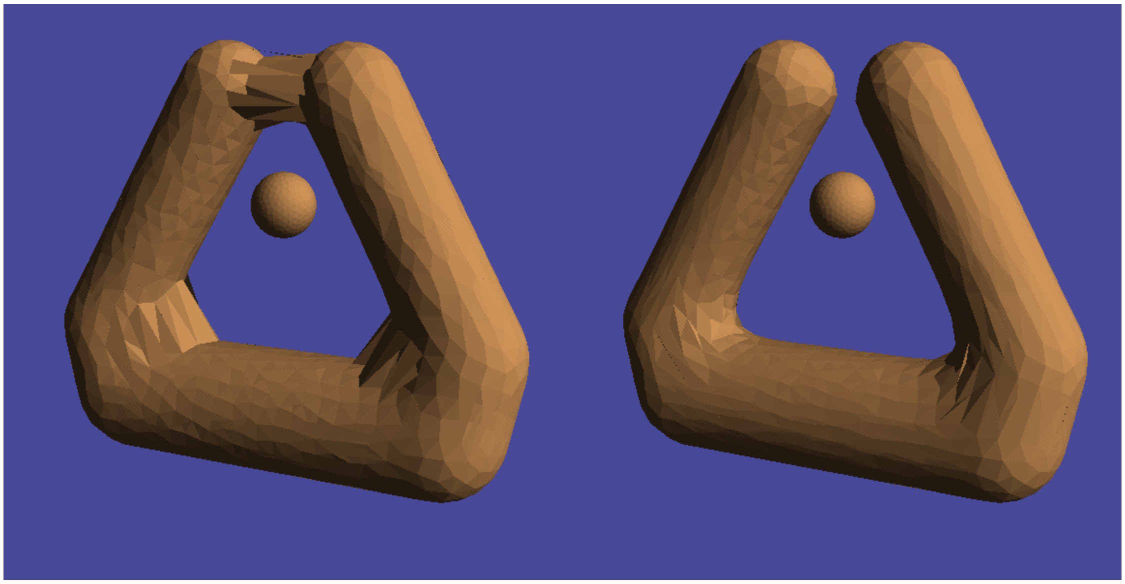

Figure 5: *The same point-set as Figure 3, first with density scaling, then with anisotropy.*

number of points	α-change	α-change with density-scaling	α-precomputation	retriangulation
360	43 msec	56 msec	1.02 sec	141 msec
600	78 msec	99 msec	2.23 sec	244 msec
965	137 msec	172 msec	3.67 sec	439 msec
2875	480 msec	570 msec	6.69 sec	1644 msec

Figure 6: *α-shape computational steps on an SGI R10K O2*

Bossen and Heckbert [2] apply anisotropy for the generation of graded triangular meshes with density varying according to external information. These can be used for applications such as finite element analysis. We believe their methods can be generalized to three dimensions using an approach similar to ours.

7. Acknowledgments

We would like to thank Ernst Mucke for making his Delaunay triangulation software available, and Hans Köhling Pedersen for making his implicit surface demonstation available on the SIGGRAPH 97 CD-ROM. We have used a modified version for generating some of our data. Finally, thanks to Neel Masters for help with system software. This work was funded in part by an NSERC PostDoctoral Fellowship.

References

1. Chandrajit L. Bajaj, Fausto Bernardini, and Guoliang Xu. Automatic reconstruction of surfaces and scalar fields from 3D scans. In Robert Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 109–118. ACM SIGGRAPH, Addison Wesley, August 1995. held in Los Angeles, California, 06-11 August 1995.
2. Frank J. Bossen and Paul S. Heckbert. A pliant method for anisotropic mesh generation. In *Proc. 5th International Meshing Roundtable*, pages 63–74, PO Box 5800, MS 0441, Albuquerque, NM, 87185-0441, 1996. Sandia National Laboratories. Also Sand. Report 96-2301.
3. H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Trans. Graph.*, 13(1):43–72, January 1994.
4. H. Edelsbrunner and N. R. Shah. Incremental topological flipping works for regular triangulations. In *Proc. 8th Annu. ACM Sympos. Comput. Geom.*, pages 43–52, 1992.
5. Hugues Hoppe, Tony DeRose, Tom Duchamp,

John McDonald, and Werner Stuetzle. Surface reconstruction from unorganized points. In Edwin E. Catmull, editor, *Computer Graphics (SIGGRAPH '92 Proceedings)*, volume 26, pages 71–78, July 1992.

6. Ernst Mucke. *Shapes and Implementations in Three-Dimensional Geometry*. PhD thesis, University of Illinois at Urbana-Champaign, 1993.

7. Shigeru Muraki. Volumetric shape description of range data using "blobby model". In Thomas W. Sederberg, editor, *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 227–235, July 1991.

8. Vaughan Pratt. Direct least-squares fitting of algebraic surfaces. In Maureen C. Stone, editor, *Computer Graphics (SIGGRAPH '87 Proceedings)*, volume 21, pages 145–152, July 1987.

9. F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, NY, 1985.

10. Barton T. Stander and John C. Hart. Guaranteeing the topology of an implicit surface polygonization for interactive modeling. In Turner Whitted, editor, *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, pages 279–286. ACM SIGGRAPH, Addison Wesley, August 1997. ISBN 0-89791-896-7.

11. Andrew P. Witkin and Paul S. Heckbert. Using particles to sample and control implicit surfaces. In Andrew Glassner, editor, *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, pages 269–278. ACM SIGGRAPH, ACM Press, July 1994. ISBN 0-89791-667-0.